REMARKS

The Non-Final Office Action mailed December 18, 2008, considered and rejected claims 1-4 and 6-22. Claims 6 and 7 were objected to because of informalities. Claims 1-4 and 6-8 were rejected under 35 U.S.C. §101 because the claimed invention was directed to non-statutory subject matter. Claims 1-4 and 6-22 were rejected under 35 U.S.C. § 102(b) as being unpatentable over Iborra et al., U.S. Publ. No. 2002/0100014 (filed Jun. 1, 2001) (hereinafter Iborra) in view of Barry Cornelius, "Comparing .NET with Java" (hereinafter Cornelius).[1]

By this response, claims 6 and 7 are amended such that claims 1–4 and 6–22 remain pending.[2] Claims 1, 11, and 15 are independent claims which remain at issue. Support for the amendments arises from dependency changes arising from a cancelled claim.[3]

As reflected in the claims, the present invention is directed generally toward data structures and methods for a type system to provide services to implement software design tools. Claim 1 recites, for instance, in combination with all the elements of the claim, a data structure encoded upon computer-readable media for a type system. The data structure comprises a ClrElement base class for capturing common functionality of objects of the type system. The data structure also includes at least one controller object in communication with the base class which validates requested services based upon a set of rules. The data structure also includes a first class which provides a level of abstraction between a second and third class.

Claim 11, in combination with all the elements of the claims, recites a method of modifying an artifact for use in a type system. The method includes receiving a request from an application programming interface to modify an artifact in the type system. The meta-model of the type system includes a ClrElement base class for capturing common functionality of objects of the type system. In response to issuing at least one instruction to a language specific controller object, a language specific controller object validates the request based on rules

---

[1] Although the prior art status of the cited art is not being challenged at this time, Applicant reserves the right to challenge the prior art status of the cited art at any appropriate time, should it arise. Accordingly, any arguments and amendments made herein should not be construed as acquiescing to any prior art status of the cited art.

[2] The amendments and remarks presented herein are consistent with the information presented by telephone by patent attorney Thomas Bonacci (reg. no. 63,368).

[3] It should be noted that the present invention and claims as recited take support from the entire Specification. As such, no particular part of the Specification should be considered separately from the entirety of the Specification.

associated with a programming language. In response to a validated request from the language specific controller, the artifact is modified.

Claim 15, in combination with all the elements of the claims, recites a method of creating an artifact for use in a type system meta-model. The method includes receiving a request from an application programming interface to create an artifact in the type system meta-model. The type system meta-model includes a ClrElement base class for capturing common functionality of objects of the type system. In response to issuing at least one instruction to a language specific controller object, the language specific controller object validates the request based on rules associated with a programming language. In response to a validated request from the language specific controller, an artifact is created.

Claims 6 and 7 were objected to because of minor informalities. The claims have now been amended to correct the typographical errors.

Claim 1 (as well as its respective dependent claims) was rejected under 35 U.S.C. § 101 as being directed toward non-statutory subject matter.[4] In particular, the Office asserted that "claim 1 recites '[a] computer-readable medium having stored thereon a data structure for a type system . . .'" and that the "'data structure' in the claim is non-functional descriptive material as there is no 'act' actually being performed . . . ."[5] The Applicants respectfully disagree and traverse the rejection. The Applicants submit that the applicable standard does, in fact, result in the claim being proper statutory subject matter.

"'*[F]unctional descriptive material*' consists of *data structures* and computer programs which impart functionality when employed as a computer component. (*The definition of 'data structure' is 'a physical or logical relationship among data elements, designed to support specific data manipulation functions.*' The New IEEE Standard Dictionary of Electrical and Electronics Terms 308 (5th ed. 1993).) 'Nonfunctional descriptive material' includes but is not limited to music, literary works, and a compilation or mere arrangement of data."[6] "*When functional descriptive material is recorded on some computer-readable medium*, it becomes structurally and functionally interrelated to the medium and *will be statutory* in most cases since use of technology permits the function of the descriptive material to be realized."[7]

---

[4] Office Comm. p. 3.
[5] Office Comm. pp. 3–4.
[6] MPEP § 2106.01 (emphasis added).
[7] MPEP § 2106.01 (emphasis added).

The Applicants submit that the data structure of claim 1 does, indeed, define "a physical or logical relationship among data elements, designed to support specific data manipulation functions."[8] I.e., the ClrElement base class supports capturing common functionality of objects of the type system. In contrast, claim 1 is not music, a literary work, or a compilation or mere arrangement of data – which are given by the MPEP as examples of non-functional descriptive material.[9]

The data structure of claim 1 defines relationships among data elements which are designed to support the data manipulation functions of the type system described in the application and recited in the claim.[10] Additionally, the data structure of claim 1 is "recorded on [a] computer-readable medium."[11] Accordingly, the elements and limitations of claim 1 do meet all the requirements for statutory subject matter as described in MPEP § 2106.01.

Further, the Office asserted, at least in partial support of the rejection, that "there is no 'act' actually being performed . . . ."[12] The Applicants respectfully submit that neither the applicable law nor MPEP § 2106.01 requires any "act actually being performed." When a data structure defines a "physical or logical relationship among data elements, designed to support specific data manipulation functions" and is recorded upon a computer-readable medium, then the standards have been met and the claim should be considered statutory under 35 U.S.C. § 101.[13]

In view of the above discussion, the Applicants respectfully submit that the rejection of claim 1 (as well as the respective dependent claims) under 35 U.S.C. § 101 was improper and should be withdrawn. The Applicants respectfully request the rejection be withdrawn and the claim(s) favorably reconsidered.

Each of the independent claims 1, 11, and 15 were rejected under 35 U.S.C. § 103(a) as being unpatentable in view of Iborra and in view of Cornelius.[14] The Applicants respectfully disagree and traverse the rejections.

---

[8] MPEP § 2106.01.
[9] *See* MPEP § 2106.01.
[10] *See* MPEP § 2106.01.
[11] MPEP § 2106.01.
[12] Office Comm. p. 4.
[13] Note that the data structure need only define "relationship[s] among data elements" and "*support* . . . data manipulation functions." *See* MPEP § 2106.01 (emphasis added). It is not required for the data structure to perform any act or function, itself. *See* MPEP § 2106.01.
[14] Office Comm. pp. 4–10.

In particular, each of the rejected independent claims recites particular and specific data structures. Claim 1 recites, in combination with the other elements of the claim, "a ClrElement base class for capturing common functionality of objects of the type system, the ClrElement base class comprising data members AttributeDeclaration, DocSummary, DocRemarks, IsEditable, IsInjected, IsCodeParseable, and IsFromReferenceAssemblies." Claim 11 recites, in combination with the other elements of the claim, "a ClrElement base class for capturing common functionality of objects of the type system, the ClrElement base class comprising data members AttributeDeclaration, DocSummary, DocRemarks, IsEditable, IsInjected, IsCodeParseable, and IsFromReferenceAssemblies." Claim 15 recites, in combination with the other elements of the claim, "the type system meta-model comprises a ClrElement base class for capturing common functionality of objects of the type system, the ClrElement base class comprising data members AttributeDeclaration, DocSummary, DocRemarks, IsEditable, IsInjected, IsCodeParseable, and IsFromReferenceAssemblies."

The Applicants submit that Iborra and Cornelius, both separately and in combination, fail to teach or suggest a ClrElement base class for capturing common functionality of objects of the type system, the ClrElement base class comprising data members AttributeDeclaration, DocSummary, DocRemarks, IsEditable, IsInjected, IsCodeParseable, and IsFromReferenceAssemblies. In its rejections, the Office did not assert that the particular ClrElement base class (i.e., data structure) was taught or suggested by the cited references. Further, the Office did not assert that the particular ClrElement comprises the particular members AttributeDeclaration, DocSummary, DocRemarks, IsEditable, IsInjected, IsCodeParseable, and IsFromReferenceAssemblies as recited and required by the claim.

In it's rejections of the claims, for teaching this particular limitation recited in the claims, the Office asserted only that

> "Cornelius . . . discloses 'Microsoft providing .Net compilers for several programming languages: managed C++, Visual Basic.Net, Jscript and C#. In addition, other people/companies are producing .NET compilers for other language including COBOL, Eiffel ... a .NET compiler writer can rely on CLR (Common Language Runtime for a larger number of task."[15]

[15] *See* Office Comm. pp. 5–6, 7–8, and 9.

In support of its assertion, the Office cited to Cornelius "page 1, last paragraph and page 2, paragraph 1."[16] The portion of Cornelius asserted to disclose these limitations reads (in its entirety):

> "2.2 The Common Language Runtime
> In the past, compiler writers have put code to support the execution of programs into a runtime system. Instead of providing a different runtime system for each programming language, the .NET Framework provides a runtime system that is used by all of the languages that are targetted at the .NET Framework. This is called the Common Language Runtime (or CLR). Code that targets the CLR is called managed code.
> Microsoft are providing .NET compilers for several programming languages: Managed C++, Visual Basic.NET, JScript and C#. In addition, other people/companies are producing .NET compilers for other languages including COBOL, Eiffel, Fortran, Haskell, ML, Perl, Python, Scheme and Smalltalk.
> A .NET compiler writer can rely on the CLR for a large number of tasks, including:
> - creating new types;
> - creating and initializing of objects;
> - tracking references to objects and providing garbage collection;
> - handling the calling of methods (including virtual methods);
> - managing the access to array elements;
> - providing support for exceptions and exception handling.
>
> All of the .NET languages have compilers that generate instructions coded in an intermediate language called MSIL (or IL). A file containing MSIL instructions can be run on any platform so long as the operating system for that platform hosts the CLR engine. Currently, a CLR engine is available for Windows XP, Windows 2000, Windows NT 4.0, Windows 98 and Windows Me.
> There is a project called Mono that is building an open-source implementation of the .NET Framework, and Microsoft are working on an implementation for FreeBSD."[17]

The Applicants submit that a "ClrElement base class comprising data members AttributeDeclaration, DocSummary, DocRemarks, IsEditable, IsInjected, IsCodeParseable, and IsFromReferenceAssemblies" is notably absent from the passage of Cornelius cited by the Office. The Applicants submit that both the cited portion of Cornelius (as well as its entirety in combination with Iborra) fails to teach or suggest a "ClrElement base class comprising data members AttributeDeclaration, DocSummary, DocRemarks, IsEditable, IsInjected, IsCodeParseable, and IsFromReferenceAssemblies."

---

[16] Office Comm. pp. 6, 8, and 9

[17] Cornelius pp. 1–2 (available at http://www.barrycornelius.com/papers/comparing.dotnet.with.java/master.pdf).

Because this limitation is recited by each of the independent claims 1, 11, and 15 and because this particular and specific limitation is neither taught nor suggested by any prior art, the Applicants submit that rejections of the independent claims under 35 U.S.C. § 103(a) as being unpatentable in view of Iborra and in view of Cornelius is improper and should be withdrawn. Accordingly, the Applicants respectfully request favorable reconsideration of independent claims 1, 11, and 15 (as well as the respective dependent claims) as they are now presented.[18]

In view of the foregoing, Applicant respectfully submits that the other rejections to the claims are now moot and do not, therefore, need to be addressed individually at this time. It will be appreciated, however, that this should not be construed as Applicant acquiescing to any of the purported teachings or assertions made in the last action regarding the cited art or the pending application, including any official notice. Instead, Applicant reserves the right to challenge any of the purported teachings or assertions made in the last action at any appropriate time in the future, should the need arise. Furthermore, to the extent that the Examiner has relied on any Official Notice, explicitly or implicitly, Applicant specifically requests that the Examiner provide references supporting the teachings officially noticed, as well as the required motivation or suggestion to combine the relied upon notice with the other art of record.

[18] The Applicants also respectfully submit that any subsequent action, if necessary, was not necessitated by any amendments to the claims and therefore respectfully request any subsequent action, if necessary, be a non-final action.

In the event that the Examiner finds remaining impediment to a prompt allowance of this application that may be clarified through a telephone interview, the Examiner is requested to contact the undersigned attorney at (801) 533-9800.

Dated this 18th day of March, 2009.

Respectfully submitted,

RICK D. NYDEGGER
Registration No. 28,651
JENS C. JENKINS
Registration No. 44,803
THOMAS M. BONACCI
Registration No. 63,368
Attorneys for Applicant
Customer No. 47973

RDN:JCJ:TMB:crb
2312431_1